ADL SCORM Assessment Guidelines

Quick Tips – Understanding `cmi.interactions`

OVERVIEW

The SCORM Data Model's `cmi.interactions` element allows a SCO to store data on the LMS about a learner's performance on an assessment. This Quick Tip provides:

- an overview of the `cmi.interactions` element.
- how to find more information in the SCORM documentation
- examples of using `cmi.interactions` to store data on the LMS

DESCRIPTION OF cmi.interactions

A single interaction describes a test item. For a single test item, a single interaction can store:

- question **identifier** – used to associate the question to a database or master list of questions
- **type** of question – multiple-choice, T/F, matching, etc
- **order of the responses** as they were presented to the learner – if the order of the responses was randomized
- **correct answer** – what the learner should have answered
- learner's **response** – what the learner actually answered
- whether the learner's response was **correct** or not

An assessment can report one interaction for every test item in the assessment. SCORM requires an LMS to be able to store 250 interactions per SCO.

An LMS can only store the information that the SCO is programmed to send to it. The information stored in `cmi.interactions` is not used by the LMS to compute a score for your SCO, so you must program the SCO to compute the score and store it using the `cmi.score.scaled` element.

Work with the instructional designer to determine the real reporting needs for your SCO. You may only need to store the details of every test item on the LMS for new assessments (to ensure that they were programmed correctly and that the assessment is properly testing the learners) or for "final exams." Collecting too much data about assessments:

- Increases your programming time (which will increase the cost to your project)
- Increases the amount of communication between the SCO and the LMS (which may slow LMS response time),
- Increases the storage requirements of the LMS (which may increase the amount of total storage you need)
- May violate the rights of learners in some cases (particularly in situations where learners are union employees).

Quick Tips – **Understanding `cmi.interactions`**

READING THE SCORM DOCUMENTATION

In the SCORM 2004, 2nd Edition documentation, the explanation of `cmi.interactions` accounts for more than 25% of the chapter on the SCORM Data Model. After reading this Quick Tip, you will understand the interactions element at a high level and be able to read the SCORM documentation to understand the details of each element under `cmi.interactions`.

`cmi.interactions.n`
Each interaction that the SCO sends to the LMS begins with `cmi.interactions.` and then an index number. The first interaction has index `0`, the second has index `1`, etc.; you **must** set interactions in order (0, 1, 2, etc.) The SCORM documentation refers to this as `cmi.interactions.n` where `n` is the index.

Where the SCORM documentation describes `cmi.interactions.n.id`, you should read this as "the `id` of the interaction." When your SCO sets the `id` for an interaction, it will replace `n` with an index, as in `cmi.interactions.0.id`.

`cmi.interactions._count`
This value contains the total number of interactions that the LMS is currently storing for this particular SCO. Use this count to determine the index for the next interaction your SCO stores on the LMS.

An overview of each element of an interaction is presented below, in the order that they are presented in the SCORM Run-Time Environment documentation. For clarity, the `cmi.interactions.n` for each element has been dropped. The elements that are in bold are used in the examples in the next section.

Element	Description
id	A unique URN within the scope of the SCO. You **must** set the id before you set anything else in the interaction. Example: "urn:adlnet.gov/assessments/q-1234"
type	The type of question. SCORM requires a unique vocabulary, so you must use one of the values provided. Example: "true-false", "choice"
objectives	The identifier of objectives related to the interaction. An LMS is required to be able to store 10 objectives per interaction. Example: "urn:adlnet.gov/objectives/obj-1234"
timestamp	The point in time that the interaction was made available to the learner. Example: "2006-06-29T03:00:00"
correct_responses	The correct response to the question. There may be multiple correct responses, particularly for fill-in-the-blank questions. Example: "1" – for a multiple-choice question Example: "true" – for a true-false question

Quick Tips – Understanding `cmi.interactions`

weighting	The weight given to this question when the SCO computed the overall score. Example: "0.5"
learner_response	The response the learner gave. Example: "2" – for a multiple-choice question Example: "false" – for a true-false question
result	Whether the answer was correct or incorrect. SCORM uses a unique vocabulary, so you must use one of the values provided. Example: "correct" or "incorrect"
latency	The time from when the interaction was made available to the learner to when the learner responded. Example: "PT5M" – 5 minutes
description	A description of the interaction, up to 250 characters. Use the description element to list the order of the response items as they were shown to the learner. Example: "2, 1, 3"

EXAMPLE USAGE

In the example from a two-question assessment below the correct answers are shown in bold. Assume the instructional designer asks you to (1) create an assessment using these two questions and (2) randomize the questions and the answers for the multiple-choice question. From other business practices, you assign the beginning of the URN to be "urn:adlnet.gov/assessments/".

Question #1234
Write the test questions after you

a. **Write the learning objectives.**
b. Develop the content.
c. Create enough information to test.

Question #4567
Avoid providing humorous distractors in a multiple choice question because they make it easier for learners to identify the correct answer.

a. **True**
b. False

Suppose that when you generate the assessment for a learner, question #4567 is presented first, and the answers for question #1234 are presented in the order b, then a, then c.

As the assessment is generated and presented to the learner, the SCO will make the following calls:

```
SetValue( "cmi.interactions.0.id",
  "urn:adlnet.gov/assessments/q-4567" );
SetValue( "cmi.interactions.0.type", "true-false" );
SetValue( "cmi.interactions.0.correct_responses.0.pattern", "true" );
SetValue( "cmi.interactions.0.description",
```

Quick Tips – **Understanding `cmi.interactions`**

```
  "response item presentation order: 0, 1" );
SetValue( "cmi.interactions.1.id",
  "urn:adlnet.gov/assessments/q-1234" );
SetValue( "cmi.interactions.1.type", "choice" );
SetValue( "cmi.interactions.1.correct_responses.0.pattern", "0" );
SetValue( "cmi.interactions.1.description",
  "response item presentation order: 1, 0, 2" );
```

Now suppose the learner responded "true" and "Developed the content." The SCO will make the following calls:

```
SetValue( "cmi.interactions.0.learner_response", "true" );
SetValue( "cmi.interactions.0.result", "correct" );
SetValue( "cmi.interactions.1.learner_response", "1" );
SetValue( "cmi.interactions.1.result", "incorrect" );
```